

无线传感器网络信息感知中的自组织算法¹⁾

王睿 梁彦 潘泉 程咏梅

(西北工业大学自动化学院 西安 710072)

(E-mail: wang.ruis@gmail.com, liangyan@nwpu.edu.cn, quanpan@nwpu.edu.cn, chengym@nwpu.edu.cn)

摘要 构造了信息获取与能量节省的综合性能指标, 通过在线优化该函数自适应地选择簇首及簇内成员, 提出了一种无线传感器网络动态协同自组织算法 (DCS), 从而实现综合性能优化意义下的网络自组织. 仿真表明: 与 IDSQ 算法相比, DCS 算法能有效避免“乒乓效应”问题, 节省通讯能量明显, 而跟踪精度相当.

关键词 无线传感器网络, 传感器协同, 信息感知, 自组织

中图分类号 TP393

A Self-Organization Algorithm in Wireless Sensor Networks

WANG Rui LIANG Yan PAN Quan CHENG Yong-Mei

(College of Automation, Northwestern Polytechnic University, Xi'an 710072)

(E-mail: wang.ruis@gmail.com, liangyan@nwpu.edu.cn, quanpan@nwpu.edu.cn, chengym@nwpu.edu.cn)

Abstract Facing the requirement of effective information acquisition and low energy cost from the wireless sensor networks, we construct an object function by combining information gain index and energy cost index for the self-organization optimization. The information index is constructed from the mutual information, so that information increment can be represented as an alternative sensor node to the sensor cluster. The energy cost is derived from the node communication hop number from the sensor node to the sensor cluster leader. By optimizing the performance index online, a dynamic collaborative self-organization (DCS) algorithm is proposed, through which the clustering leader can be automatically elected and its member nodes can be selected sequentially. Compared with IDSQ in our simulation, DCS can relieve the problem of “too frequent leader switches” and achieve almost the same tracking accuracy at the cost of low energy consumption.

Key words Wireless sensor networks, sensor collaboration, information acquirement, self-organization

1 引言

随着科学技术的进步, 尤其是微电子技术、通讯技术及计算技术的飞速发展, 无线传感器网络 (WSN) 成为目前的一个研究热点^[1]. WSN 是由大量廉价、智能化的微传感器节点通过无线连接组成的复杂网络系统, 其主要功能是智能感知外界环境, 并将处理后的信息提供给用户. WSN 在信息感知中两个重要的性能指标是精确的信息感知能力和较低的能量消耗. 这两个指标常相互矛盾. 利用自组织技术提高网络节点信息探测与处理的协同能力是解决二者矛盾的有效途径.

1) 国家自然科学基金项目 (60404011, 60372085) 资助

Supported by National Natural Science Foundation of P. R. China (60404011, 60372085)

收稿日期 2005-5-30 收修改稿日期 2006-3-7

Received May 30, 2005; in revised form March 7, 2006

文献 [2] 提出 IDSQ 方法, 用于 WSN 中时序节点的协同组织问题, 即在一定的计算、通讯资源下, 按照时序选取参与信息获取与处理的节点, 以达到信息获取的最大化. 该方法实际上是节点自组织问题中的簇首选取方法, 但并没有过多考虑簇内其余节点的生成方法; 而且在某些情况下会出现簇首在几个固定节点间频繁切换的情况, 即所谓的“乒乓效应”.

本文提出动态协同自组织 (Dynamic collaborative self-organization, DCS) 方法, 依据系统的能量消耗与信息收益建立综合性能指标, 由性能指标函数动态选择簇首及生成簇内节点, 一方面使得每一时刻信息感知计算只局限于簇内进行, 另一方面又保证当前生成簇在获得足够信息量的基础上消耗较少的能量, 从而使系统获得良好的综合性能.

2 动态协同自组织算法

在 WSN 中, 进行第 $t-1$ 时刻观测后, 本拍簇首以外的其他节点都处于休眠状态, 簇首节点可由信息融合结果做出 t 时刻的状态预测 $\hat{x}_{t|t-1}$, 根据预测值及性能指标可以判断是否需要切换簇首以及如何恰当的选取簇内成员, 进行动态自组织, 当信息收益函数达到阈值, 宣告自组织完毕, 并由簇首唤醒簇内相关节点进行探测, 探测结果发回簇首, 进行集中式信息融合处理并保存相关结果, 而簇内成员重回休眠状态. 本算法所要解决的就是如何在 WSN 中自适应的选取簇首和动态组织节点, 以平衡感知能力与能量节省间的矛盾.

2.1 性能指标函数设计

首先建立性能指标函数 $O_t^a(k) = f(\varphi_{utility}(p(x_t|\{z_t^i\}_{i \in G_t^a} \cup \{z_t^k\}), \psi_{cost}(a, k)))$, 其中 a, k 分别表示该时刻的簇首和备选成员节点; $\varphi_{utility}(p(x_i|\{z_t^i\}_{i \in G_t^a} \cup \{z_t^k\}))$ 表示当前节点 k 加入簇 G_t^a 后带来的信息收益, 信息收益越大说明其带来的信息量越多; 而 $\psi_{cost}(a, k)$ 表示该节点加入当前簇带来的能量消耗. $O_t^a(k)$ 表示 k 节点加入当前簇后, 综合考虑信息收益与能量消耗的综合性能指标, 它随着 $\varphi_{utility}$ 的增大而增大, 随着 ψ_{cost} 的增大而减小.

本文设计性能指标函数为如下形式:

$$f(\varphi_{utility}, \psi_{cost}) = \varphi_{utility} / \psi_{cost} \quad (1)$$

利用互信息来表示信息收益函数 $\varphi_{utility}$, 即:

$$\begin{aligned} \varphi_{utility}(p(x_{t+1}|\{z_{t+1}^i\}_{i \in G_t^a} \cup \{z_{t+1}^k\})) &= I_k(\hat{x}_{t+1}; \hat{z}_{t+1}^k | z^t) = \\ E_{p(\hat{x}_{t+1}, \hat{z}_{t+1}^k | z^t)}[\log p(\hat{x}_{t+1}, \hat{z}_{t+1}^k | z^t) / (p(\hat{x}_{t+1} | z^t) \times p(\hat{z}_{t+1}^k | z^t))] \end{aligned} \quad (2)$$

由互信息定义可知, $I_k(\hat{x}_{t+1}; \hat{z}_{t+1}^k | z^t)$ 越大, 表示在已知 z^t 条件下, k 节点的量测 \hat{z}_{t+1}^k 提供的有关状态估计 \hat{x}_{t+1} 的信息量越多, 符合我们对信息收益函数的要求.

WSN 的通讯消耗了绝大多数的能量, 因此代价函数 $\psi_{cost}(a, k)$ 表示为:

$$\psi_{cost}(a, k) = e^{hop(a, k)} \quad (3)$$

其中 $hop(a, k)$ 表示由当前簇首 a 到节点 k 的通讯跳数. 公式 (3) 表明随着节点到簇首的通讯跳数增加, 其代价迅速增大.

由 (2) 和 (3) 式可得如下的性能指标函数:

$$O_t^a(k) = E_{p(\hat{x}_{t+1}, \hat{z}_{t+1}^k | z^t)}[\log \frac{p(\hat{x}_{t+1}, \hat{z}_{t+1}^k | z^t)}{p(\hat{x}_{t+1} | z^t) \cdot p(\hat{z}_{t+1}^k | z^t)}] / e^{hop(a, k)} \quad (4)$$

(4) 式各概率函数求解如下:

$$p(\hat{x}_{t+1}, \hat{z}_{t+1}^k | z^t) = p(\hat{x}_{t+1} | z^t) \times p(\hat{z}_{t+1}^k | \hat{x}_{t+1}), \quad p(\hat{z}_{t+1}^k | z^t) = \sum_{\hat{x}_{t+1} \in X} p(\hat{x}_{t+1}, \hat{z}_{t+1}^k | z^t)$$

其中 X 为状态空间集合.

2.2 算法设计

定义以下判断标志变量:

$$\theta_0 = \begin{cases} 0, & \text{簇首切换} \\ 1, & \text{簇首不切换} \end{cases}, \theta_1 = \begin{cases} 0, & \text{簇自组织完毕} \\ 1, & \text{簇自组织未完} \end{cases}, \\ \theta_2 = \begin{cases} 0, & \text{集合为空} \\ 1, & \text{集合不为空} \end{cases}, \theta_3 = \begin{cases} 0, & \text{簇首已切换} \\ 1, & \text{簇首未切换} \end{cases}$$

系统初始输入包括: 初始时刻及该时刻相关信息感知结果.

1) 初始化 $n_0 = d_0$, $n_1 = d_1$, $a_t = a_{t-1}$, $\theta_3 = 1$. n_0, n_1 分别为簇首及簇内节点备选集合的选择半径并赋初值 d_0, d_1 , 该值与节点的有效通讯半径有关. a_t 和 a_{t-1} 表示 t 时刻及 $t-1$ 时刻簇首.

2) 判断簇首切换

$$\theta_0 = \begin{cases} 0, & \varphi_{utility}(p(x_t|Z^{t-1}, z_t^{a_t})) \leq M_0 \\ 1, & \varphi_{utility}(p(x_t|Z^{t-1}, z_t^{a_t})) > M_0 \end{cases} \quad (5)$$

M_0 为簇首信息收益的接受阈值, $\theta_0 = 1$ 表示当前簇首可以接受, $\theta_0 = 0$ 表示当前簇首需切换. 当 $\theta_0 = 1$ 时进入下一步, 进行簇生成过程; 当 $\theta_0 = 0$ 时转至 1), 进入簇首切换.

3) 建立 t 时刻初始簇集合 $G_t^a = \{a_t\}$, 该集合元素仅包含当前簇首 a_t .

4) 确定 t 时刻簇内节点备选集合 $V_t = \{k|k \notin G_t^a, L(k, a_t) < n_1\}$, G_t^a 为簇集合, $L(k, a_t)$ 为节点 k 与簇首 a_t 间的有效通讯距离.

5) 选择具有最大性能指标增量的节点 $\hat{j} = \arg \max_{j \in V_t} (O_t^a(j))$

6) 重新判断簇首

$$\theta_0 = \begin{cases} 0, & O_t^a(\hat{j}) \leq M_1 \\ 1, & O_t^a(\hat{j}) > M_1 \end{cases} \quad (6)$$

M_1 为簇内节点性能指标的接受阈值. 当节点满足性能指标阈值时, $\theta_0 = 1$, 进入下一步; 当节点不能满足时, 则应重新选择簇首, $\theta_0 = 0$, 转至 1), 进入簇首切换.

7) 更新: $G_t^a = G_t^a \cup \hat{j}$, $V_t = V_t - \hat{j}$

8) 判断自组织是否完毕

$$\theta_1 = \begin{cases} 0, & \varphi_{utility}(p(x_t|Z^{t-1}, \{z_t^i\}_{i \in G_t^a})) \geq M_2 \\ 1, & \varphi_{utility}(p(x_t|Z^{t-1}, \{z_t^i\}_{i \in G_t^a})) < M_2 \end{cases} \quad (7)$$

M_2 为簇集合的信息收益接受阈值, $\theta_1 = 0$ 表示该簇集合已产生足够信息收益, 可以结束自组织, $\theta_1 = 1$ 表示还需继续加入新的节点. $\theta_1 = 1$ 时进入下一步; $\theta_1 = 0$ 时转至 12), 进入协同感知.

9) 判断集合是否为空

$$\theta_2 = \begin{cases} 0, & V_t = \phi \\ 1, & V_t \neq \phi \end{cases} \quad (8)$$

$\theta_2 = 0$ 表示备选集合为空, $\theta_2 = 1$ 表示备选集合不为空. $\theta_2 = 0$ 时进入下一步; $\theta_2 = 1$ 时跳转至 5), 在该集合内继续选取节点.

10) 扩大备选范围: $n_1 = n_1 + d$. d 表示扩大备选集合的步长参数. 然后跳转回 4).

11) 簇首切换: $\theta_3 = 0$ 表示簇首已切换过, $\theta_3 = 1$ 表示簇首还未切换. $\theta_3 = 0$ 时, 算法结束; $\theta_3 = 1$ 时, 选取具有最大信息收益的节点 a'_t 如下:

$$a'_t = \arg \max_{j \in V'_t} (\varphi_{utility}(p(x_t | Z^{t-1}, z_t^j))) \quad (9)$$

$V'_t = \{j | L(j, \hat{x}_t) < n_0\}$, \hat{x}_t 为状态预测, n_0 为簇首选择半径参数. $a_t = a'_t$, $\theta_3 = 0$, 返回 2).

12) 协同感知: 当前簇首收集簇内节点量测信息, 进行信息融合处理, 得出信息感知结果.

13) 循环: 保存相关信息, $t = t + 1$, 返回 1).

3 仿真分析

考虑算法仿真环境为 WSN 的目标跟踪. 目标最大速度为 20/ 采样周期, 运动模式选择为 RWP 模型. 监视区域为 (0, 0) 到 (300, 120) 的矩形区域, 在其中随机散布 70 个传感器节点. 节点间有效通讯半径为 40. 传感器模型^[3] 为 $z_k = A / \|x - s_k\|_2 + \omega_k$, 其中 k 表示传感器编号, z 为量测值, A 为目标能量辐射强度, x 为目标当前位置, s 为传感器节点位置, 量测误差 ω 服从 $N(0, 0.05^2)$, 由于被观测对象状态的多变性及环境噪声干扰, A 为 [0,80] 均匀分布的随机变量. 我们采用贝叶斯滤波^[4] 进行非线性滤波. 对状态空间进行 5×5 方格大小的离散化处理. 仿真采用与 [3] 相同的条件, 分别对 DCS 与 IDSQ 算法进行 50 次 Monte Carlo 仿真. 算法参数取值标称点为 $[M_0 \ M_1 \ M_2] = [0.08 \ 0.003 \ 0.2]$.

本文利用均方根误差 (RMSE) 来表示算法获得信息量的多少. 其依据是: 由信息不等式, 无偏估计的方差下限是 Fisher 信息矩阵的逆, 而贝叶斯滤波一般能保证估计的无偏性, 因此 RMSE 能近似作为衡量算法获得信息量的指标. 算法所消耗的能量则以节点间通讯跳数来表示. 此外, 我们记录了算法中每拍簇首所在的节点位置, 以展现簇首切换.

图 1 为被观测对象的运动轨迹, 其中细实线即为本次仿真所生成的 100 个采样周期的运动轨迹, 虚线表示利用 DCS 算法并通过贝叶斯滤波得到的轨迹. 图 2 为 DCS、IDSQ 两种算法的均方根误差比较, 结果表明虽然 IDSQ 算法利用了所有邻居节点的量测信息, DCS 算法只利用了部分具有较高信息量节点信息, 但是 DCS 算法仅损失了少量的信息, 这主要反映两者在 RMSE 指标差异不大. 图 3 表示在单次仿真中簇首所在的节点, 横坐标为仿真拍数, 纵坐标为传感器节点的编号, 从图中可见 IDSQ 算法簇首切换相当频繁, 在 40~50、80~90 等时间段处可见明显的“乒乓效应”, 而 DCS 算法的簇首切换相对平缓得多, 这说明 DCS 算法有效降低了“乒乓效应”发生的可能性. 图 4 为两种算法在通讯跳数方面的比较. 从图中可以看出就能耗指标通讯跳数而言, IDSQ 算法比 DCS 要大得多, 分析其原因, IDSQ 算法在选取簇内成员节点时不加区分, 使大量只有很小信息收益的节点也参与信息感知, 结果耗费了巨大的通讯能量, 而 DCS 算法则更有效率, 根据性能指标函数有针对性的选取节点加入该簇, 当达到一定指标以后停止加入新的节点, 以损失极少的信息量换取了大量的能量. IDSQ 算法由于需要频繁切换簇首, 而簇首在进行交接的时候需要传递大量数据, 耗费了可观的通讯能量, 由于 DCS 算法降低了“乒乓效应”发生的可能性, 也就有效减少了这部分能量开销.

此外, 我们对阈值 M_0 , M_1 , M_2 的鲁棒性进行了仿真研究. 结果表明: 阈值在标称值附近变化时, 算法性能总体接近. 限于篇幅, 仿真结果图略.

4 结论

本文提出了一种用于 WSN 信息感知的动态协同自组织方法. 该方法基于信息有效获取与能量合理利用建立性能指标函数, 动态进行簇首节点的确定和簇内节点的组成. 该方

法通过设定簇首切换阈值而有效避免了“乒乓效应”,通过建立簇内成员选取准则提高了自组织中的高效性. 仿真结果表明了算法的有效性.

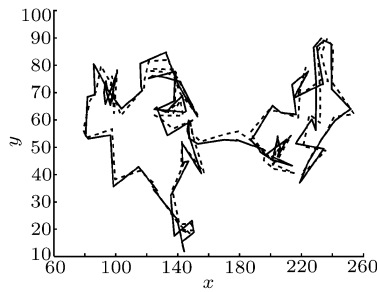


图 1 被观测对象的运动轨迹

Fig. 1 Trajectory of target

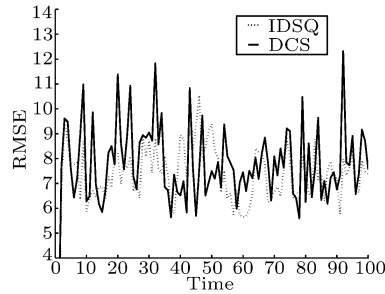


图 2 两种算法的均方根误差比较

Fig. 2 Comparison of RMSE results

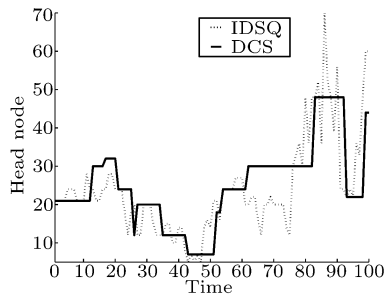


图 3 两种算法簇首切换对比

Fig. 3 Comparison of cluster head switched

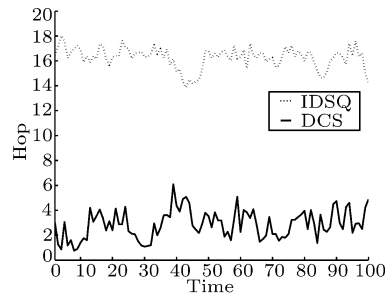


图 4 两种算法通讯跳数比较

Fig. 4 Comparison of communication hop

References

- 1 Ian F Akyildiz, Wang X D, Wang W L. Wireless mesh networks: a survey. *Computer Networks*, 2005, **47**(4): 445~487
- 2 Feng Z, Jaewon Shin, Reich J. Information-driven dynamic sensor collaboration. *Signal Processing Magazine, IEEE*, 2002, **19**(2): 61~72
- 3 Liu J, Reich J, Zhao F. Collaborative in-network processing for target tracking. *Journal on Applied Signal Processing (EURASIP)*, 2003, **23**(4): 378~391
- 4 Lawrence D Stone, Carl A Barlow, Thomas L. Corwin. Bayesian Multiple Target Tracking. Boston, London: Artech House, 1999. 29~53

王 睿 西北工业大学自动化学院博士生. 研究方向为无线传感器网络、群体智能和信息融合.

(WANG Rui Ph.D. candidate in College of Automation at Northwestern Polytechnic University. His research interests include wireless sensor networks, swarm intelligence, and information fusion.)

梁 彦 西北工业大学自动化学院副教授. 研究方向为信息融合、无线传感器网络.

(LIANG Yan Associate professor in College of Automation at Northwestern Polytechnic University. His research interests include information fusion and wireless sensor networks.)

潘 泉 西北工业大学自动化学院教授. 研究方向为信息融合、多目标跟踪、智能处理.

(PAN Quan Professor in College of Automation at Northwestern Polytechnic University. His research interests include information fusion, target tracking, and intelligent processing.)

程咏梅 西北工业大学自动化学院教授. 研究方向为人工智能、信息融合、信号处理.

(CHENG Yong-Mei Professor in College of Automation at Northwestern Polytechnic University. Her research interests include artificial intelligence, information fusion, and signal processing.)